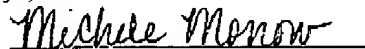


Docket No. AUS9-2000-0449-US1

PATENT**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**In re application of: **Kehne et al.**Serial No. **09/726,290**Filed: **November 30, 2000****For: Method and Apparatus for
Updating New Versions of Firmware
in the Background**§
§
§
§
§
§
§
§
§Group Art Unit: **2122**Examiner: **Rutten, James D.****Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450****Certificate of Transmission Under 37 C.F.R. 51.3(a)**

I hereby certify this correspondence is being transmitted via facsimile to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, facsimile number (703) 872-9306 on February 2, 2005.

By:


Michele Morrow**APPEAL BRIEF (37 C.F.R. 41.37)**

This brief is in furtherance of the Notice of Appeal, filed in this case on August 10, 2004, and of the Notification of Non-Compliant Appeal Brief dated January 4, 2005.

The fees required under § 41.20(B)(2), and any required petition for extension of time for filing this brief and fees therefore, are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

(Appeal Brief Page 1 of 26)
Kehne et al. - 09/726,290

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-36

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: NONE
2. Claims withdrawn from consideration but not canceled: NONE
3. Claims pending: 1-36
4. Claims allowed: NONE
5. Claims rejected: 1-36

C. CLAIMS ON APPEAL

The claims on appeal are: 1-36

STATUS OF AMENDMENTS

An Amendment after Final Rejection was not filed. Therefore, claims 1-36 on appeal herein are as amended in the Response to Office Action filed March 11, 2004.

SUMMARY OF CLAIMED SUBJECT MATTER

The present invention, as recited in claim 1 on Appeal, is directed to a method for updating system firmware in a system component within a data processing system. The invention may be implemented in a data processing system such as data processing system 100 illustrated in Figure 1, and described from page 5, line 4 to page 7, line 1. As best illustrated in Figure 2, System Power Control Network (SPCN) cards 220-226 are provided in input/output drawers 210-216. Each SPCN card 220-226 contains a SPCN flash memory 230-236 that contains SPCN firmware (page 8, lines 12-16) to be updated with new SPCA firmware 204 stored in flash memory 202 (page 8, lines 5-10).

As described on page 2, lines 5-12 of the specification:

This SPCN firmware typically takes 45 minutes to complete for a four (4) drawer computer system. If there are more drawers in the system, then it will take longer than 45 minutes to update. Currently, the operating system cannot be loaded until this firmware update is finished. Thus, the user may have to wait an hour or more before the system is usable.

In accordance with the present invention, accordingly, updating firmware, such as SPCN firmware, is delayed until a notification is received that control has been transferred to a host operating system or until an operating system has been loaded, following completion of an initialization procedure. With the present invention, therefore, the service processor and the SPCN cards are available to assist the system firmware in the initialization procedure. After control has been transferred to a host operating system or after the operating system has been loaded, following completion of the initialization procedure, the service processor is no longer needed by the system firmware, and may then be used to update the SPCN firmware in the background while the data processing system is available to a user for other actions.

As further recited in claim 1, in response to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure (Page 12, lines 18-21, Step 310 in Figure 3), a service processor 201 (Figure 2) determines whether the system component, the SPCN cards, has a current level of the firmware (Page 12, line 27-Page 13, line 2, Steps 312-316 in Figure 3). In response to a determination that the system component does not have the current level of the firmware, a copy of the firmware stored in the

system component is updated with the new SPCA firmware 204 (Page 13, lines 6-10, No output of Step 316 in Figure 3). The updating is performed in a background operation while the data processing system remains available to a user for other actions (Page 12, lines 18-26).

Independent claim 7 is also directed to a method for updating system firmware in a data processing system. As recited in claim 7, in the background, and responsive to receiving a notification that an operating system has been loaded following completion of an initialization procedure (Page 12, lines 18-21, Step 310 in Figure 3), a determination is made whether a level of a firmware copy on a system component matches a current level of firmware stored on a non-volatile memory within the system (Page 12, line 27-Page 13, line 2, Steps 312-316 in Figure 3). Responsive to a determination that the level of the firmware copy is different from the current level (Page 13, lines 6-10, No output of Step 316 in Figure 3), transferring the current level of firmware to the system component to update the firmware copy on the system component.

Independent claims 13 and 19 are directed to a computer program product in a computer readable medium for updating firmware in a system component. Independent claim 13 is a computer program product claim counterpart to claim 1, and independent claim 19 is a computer program product claim counterpart to claim 7.

Independent claims 25 and 31 are directed to a system for updating firmware in a system component. Independent claim 25 is a system claim counterpart to claim 1, and independent claim 31 is a system claim counterpart to claim 7. In claims 25 and 31, the first means and the second means correspond to system processor 201 (page 12, line 18 to page 13, line 21).

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

1. Whether claims 1, 3, 4, 6, 7, 10, 12, 13, 15, 16, 18, 19, 22, 24, 25, 27, 28, 30, 31, 34 and 36 are unpatentable under 35 U.S.C. 103(a) over Appellants' admission of prior art on pages 1 and 2 of the specification (APA), in view of U.S. Patent No. 5,822,692 (Krishan) and U.S. Patent No. 5,339,450 (Nagahara).

2. Whether claims 2, 8, 14, 20, 26 and 32 are unpatentable under 35 U.S.C. 103(a) over Appellants' admission of prior art on pages 1 and 2 of the specification (APA), in view of U.S. Patent No. 5,822,692 (Krishan), U.S. Patent No. 5,339,450 (Nagahara) and U.S. Patent No. 6,357,021 (Kitigawa).

3. Whether claims 5, 11, 17, 23, 29 and 35 are unpatentable under 35 U.S.C. 103(a) over Appellants' admission of prior art on pages 1 and 2 of the specification (APA), in view of U.S. Patent No. 5,822,692 (Krishan), U.S. Patent No. 5,339,450 (Nagahara) and "Programming Embedded Systems in C and C++" by Michael Barr and published in January, 1999 (Barr).

4. Whether claims 9, 21 and 33 are unpatentable under 35 U.S.C. 103(a) over Appellants' admission of prior art on pages 1 and 2 of the specification (APA), in view of U.S. Patent No. 5,822,692 (Krishan), U.S. Patent No. 5,339,450 (Nagahara), U.S. Patent No. 6,357,021 (Kitigawa), and "Computer Users Dictionary" published by Microsoft Press in 1998 (Microsoft).

ARGUMENT

I. 35 U.S.C. § 103, Obviousness, Claims 1, 3, 4, 6, 7, 10, 12, 13, 15, 16, 18, 19, 22, 24, 25, 27, 28, 30, 31, 34 and 36

The Examiner has rejected claims 1, 3, 4, 6, 7, 10, 12, 13, 15, 16, 18, 19, 22, 24, 25, 27, 28, 30, 31, 34 and 36 under 35 U.S.C. § 103(a) as being unpatentable over Appellants' admission of prior art on pages 1 and 2 of the specification (APA), in view of U.S. Patent No. 5,822,692 (Krishan) and U.S. Patent No. 5,339,450 (Nagahara).

In rejecting the claims as being unpatentable over APA in view of Krishan and Nagahara, the Examiner states:

As per claim 1, APA discloses:

determining, by a service processor, whether a system component has a current level of the firmware (page 1 line 32 - page 2 line 5: "At this time, the SPCN task will read the SPCN firmware level (i.e. version) on the service processor flash. If that firmware level does not match with the level of firmware on the SPCN card, then the SPCN task will transmit a new SPCN firmware image to the SPCN card while the OS is running." The SPCN task is run by the service processor.); and

responsive to a determination that the system component does not have the current level of the firmware, updating a copy of the firmware stored in the system component (page 2 lines 2-5 as cited above).

APA does not disclose *receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure or updating in a background operation while the data processing system remains available to a user for other actions.*

However, in an analogous environment, Nagahara teaches sending notification to a service processor (SVP) that an operating system is running and initialized (column 8 lines 21-27: "The input/output instruction for the operating system message (this instruction notifies that the operating system is stored in the main storage unit 2 to the SVP terminal 10 and moreover sequentially notifies the executing condition of OS to an operator) is transferred to the host command program 324 of SVP 3 through the multiplexer channel and CIA 37 of SVP 3.").

Also in an analogous environment, Krishnan teaches updating the firmware on a PCMCIA device that operates under control of an operating system (column 16 lines 8-10: "If the host has a newer version of the firmware, it transmits the newer version to data communication device 10 via TDX line 140.") The operation of transmitting is inherently performed under the control of a host operating system that conforms to the PCMCIA standard. Furthermore, operation of a PCMCIA device inherently allows the device to be powered on, powered off, and updated in a background operation of the host operating system according the PCMCIA standard (note that further information

regarding PCMCIA technology can be found in "An Introduction to PCMCIA and PC Card Technology" by Synchrotech).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use the firmware update method of APA with Nagahara's notification signal and Krishnan's background update. One of ordinary skill in the art would have been motivated to update the firmware of a device to correct software bugs in a manner that is convenient with minimal downtime of the system. One would be further motivated to routinely and automatically install available firmware upgrades to ensure correction of bugs.

Final Office Action dated June 4, 2004, pages 4 - 5.

Independent claim 1 on appeal herein reads as follows:

1. A method of updating firmware in a system component within a data processing system, the method comprising:
responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure, determining, by a service processor, whether the system component has a current level of the firmware; and
responsive to a determination that the system component does not have the current level of the firmware, updating a copy of the firmware stored in the system component in a background operation while the data processing system remains available to a user for other actions.

Appellants submit that neither APA, Nagahara and Krishan, nor the combination of APA, Nagahara and Krishan, discloses or suggests the step of "responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure, determining, by a service processor, whether the system component has a current level of the firmware", and then updating a copy of the firmware stored in the system component in a background operation "responsive to a determination that the system component does not have the current level of the firmware". Appellants submit, accordingly, that claim 1 is not obvious in view of the references and should be allowable in its present form.

APA, on pages 1 and 2 of the present application describes a known method for updating System Power Control Network (SPCN) firmware in a data processing system. As indicated above, the Examiner acknowledges that APA does not disclose "receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure or updating in a background operation while the data processing system remains

available to a user for other actions". The Examiner contends, however, referring to column 8 lines 21-27 of Nagahara, that Nagahara teaches sending notification to a service processor (SVP) that an operating system is running and initialized; and that Krishan teaches, in column 16, lines 8-10, updating firmware on a PCMCIA device that operates under control of an operating system. The Examiner then concludes that it would be obvious "to use the firmware update method of APA with Nagahara's notification signal and Krishan's background update". Appellants respectfully disagree.

Nagahara is directed to a system that includes a plurality of terminals that are able to function both as service processor terminals and work station terminals. Nagahara recognizes that service processor terminals of a computer system are often installed in a computer room separate from work stations. Accordingly, if a system failure occurs, an operator must go to the computer room and conduct initial program loading by operating a service processor terminal to re-initialize the computer. Nagahara solves this problem by enabling a terminal to function both as a work station terminal and as a service processor terminal. **Figures 3a and 3b** in Nagahara illustrate the operation of programs when a computer system has gone down as a result of a program error in the main storage unit. Step 128 of the method is described at Column 8, lines 21-27 of Nagahara as follows:

Step 128

The input/output instructions for the operating system message (this instruction notifies that the operating system is stored in the main storage unit 2 to the SVP terminal 10 and moreover sequentially notifies the executing condition of OS to an operator) is transferred to the host command program 324 of SVP 3 through the multiplexer channel and CIA 37 of SVP 3.

Nagahara discloses only that a service processor terminal is notified when an operating system has been stored in a main memory, and that an operator is notified of the executing condition of the operating system. These actions, however, are in conjunction with re-initializing a computer following a system failure and are not related to "determining, by a service processor, whether the system component has a current level of the firmware" in response to receiving a notification that control has been transferred to a host operating system following an initialization procedure, and then updating a copy of the firmware stored in the system

component responsive to a determination that the component does not have the current level of firmware.

Krishan is directed to a data communication device, such as a pager or e-mail receiver, that can be connected to a host computer. Among other features of the data communication device, Krishan discusses a procedure by which firmware of the device can be updated. In particular, in column 16, lines 14-33, Krishan recites:

Referring again to FIG. 4C, in normal operation radio frequency controller circuit 80 will execute the instructions loaded into program memory 81, while data memory 82 is used to store data. However, during the process of updating the firmware, radio frequency controller circuit 80 toggles multiplexer 84, to switch ROM_OE line 179 and RAM_OE line 180 and to switch terminals 181 and 182. As a result, radio frequency controller circuit 80 temporarily executes the instructions in data memory 82, rather than in program memory 81. Updating instructions in data memory 82, which were transmitted with the new firmware, direct radio frequency controller circuit 80 to load the new firmware from data memory 82 into program memory 81, thereby updating the firmware for radio frequency controller 46. After program memory 81 has been updated, the updating instructions in data memory 82 direct radio frequency controller circuit 80 to run a reinitiation routine, during which multiplexer 84 is returned to its initial setting. After running the reinitiation routine, the updating of the firmware of radio frequency circuitry 36 is complete.

In Krishan, it is an objective to update firmware in a data communication device by, for example, downloading the updated firmware from a host. The reference does not discuss any conditions that must be satisfied before updating can occur. In the present invention, on the other hand, as recited in claim 1, firmware in a data processing system is updated "responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure".

A fundamental notion of patent law is the concept that invention lies in the new combination of old elements. Therefore, a rule that every invention could be rejected as obvious by merely locating each element of the invention in the prior art and combining the references to formulate an obviousness rejection is inconsistent with the very nature of "invention." Consequently, a rule exists that a combination of references made to establish a *prima facie* case of obviousness must be supported by some teaching, suggestion, or incentive contained in the prior art which would have led one of ordinary skill in the art to make the claimed invention.

The Examiner bears the burden of establishing a *prima facie* case of obviousness based on the prior art when rejecting claims under 35 U.S.C. § 103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). The requirements for establishing a *prima facie* case of obviousness in view of a combination of references are set forth in detail in Section 2142 of the MPEP and include the requirements that the Examiner explain in detail why the combination of the teachings is proper, that the Examiner provide a clear and convincing line of reasoning as to why an artisan would have found the claimed invention obvious in light of the teachings of the references, and that the Examiner provide a showing that it is the prior art and not the Applicant's own disclosure that teaches the combination asserted by the Examiner.

Appellants submit that the Examiner has not fulfilled the burden of establishing a *prima facie* case of obviousness of claim 1 based on the cited prior art. In particular, Appellants submit that it would not be obvious to one of ordinary skill in the art, in view of the separate teachings of APA, Nagahara and Krishan, to combine the references to teach "responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure, determining, by a service processor, whether the system component has a current level of the firmware", and then updating the firmware responsive to a determination that the system component does not have the current level of the firmware. Although Nagahara may disclose notifying a service processor terminal when an operating system has been loaded, Nagahara does not disclose or suggest "determining, by a service processor, whether the system component has a current level of the firmware" responsive to receiving such a notification. Although Krishan may disclose updating firmware in a data communication device by, for example, downloading the updated firmware from a host, Krishan does not, in any way, discuss any conditions that must be satisfied before updating can occur.

Appellants submit that the Examiner has taken separate and unrelated teachings from APA, Nagahara and Krishan and combined the separate teachings in an effort to achieve the present invention. There is, however, no teaching, suggestion, or incentive in any of the three references which would have led one of ordinary skill in the art to combine the references as proposed by the Examiner. Only the present application contains any suggestion for combining the references as proposed by the Examiner. Appellants submit, therefore, that the rejection is based on hindsight using Appellants' own disclosure as a guide, and that the Examiner,

therefore, has not satisfied the requirements for establishing a *prima facie* case of obviousness in view of a combination of references.

Independent claim 1, accordingly, is believed to patentably distinguish over the references and to be allowable in its present form.

Claims 3, 4 and 6 depend from and further restrict claim 1, and should also be allowable in their present form, at least by virtue of their dependency.

Independent claims 13 and 25 have been amended in a manner similar to claim 1, and should also be allowable in their present form, together with claims 15, 16, 18, 27, 28 and 30 depending therefrom.

Independent claim 7 reads as follows:

7. A method for updating system firmware in a data processing system, the method comprising:

in the background, and responsive to receiving a notification that an operating system has been loaded following completion of an initialization procedure, determining whether a level of a firmware copy on a system component matches a current level of firmware stored on a non-volatile memory within the system; and

responsive to a determination that the level of the firmware copy is different from the current level, transferring the current level of firmware to the system component to update the firmware copy on the system component.

For similar reasons as discussed above with respect to claim 1, neither APA, Nagahara nor Krishan discloses or suggests the step of "responsive to receiving a notification that an operating system has been loaded following completion of an initialization procedure, determining whether a level of a firmware copy on a system component matches a current level of firmware stored on a non-volatile memory within the system", and transferring the current level of firmware to the system component to update the firmware copy on the system component "responsive to a determination that the level of the firmware copy is different from the current level of the firmware copy". Accordingly, claim 7 should also be allowable in its present form, together with claims 10 and 12 dependent thereon.

Independent claims 19 and 31 have been amended in a manner similar to claim 7, and should also be allowable in their present form, together with claims 22, 24, 34 and 36 dependent thereon.

II. 35 U.S.C. § 103, Obviousness, Claims 2, 8, 14, 20, 26 and 32

Claims 2, 8, 14, 20, 26 and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over APA, Krishan, and Nagahara, and further in view of U.S. Patent No. 6,357,021 to Kitagawa et al. (hereinafter "Kitagawa"). Kitagawa is cited as disclosing notifying a user of a firmware update failure.

Kitagawa, however, does not supply the deficiencies in APA, Krishan and Nagahara as discussed above, and claims 2, 8, 14, 20, 26 and 32 should be allowable in their present form, at least, by virtue of their dependency from allowable claims, and it is respectfully requested that the Board reverse the Examiner's Final Rejection of those claims.

III. 35 U.S.C. § 103, Obviousness, Claims 5, 11, 17, 23, 29 and 35

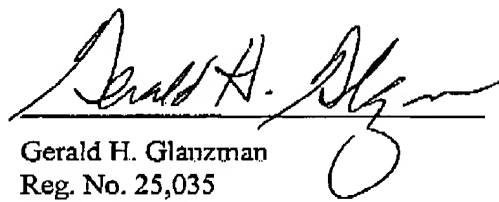
Claims 5, 11, 17, 23, 29 and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over APA, Krishan and Nagahara, and further in view of "Programming Embedded Systems in C and C++" by Michael Barr (hereinafter "Barr"). Barr is cited as disclosing a non-volatile random access memory to store data.

Barr, however, does not supply the deficiencies in APA, Krishan and Nagahara as discussed above, and claims 5, 11, 17, 23, 29 and 35 should be allowable in their present form, at least, by virtue of their dependency from allowable claims, and it is respectfully requested that the Board reverse the Examiner's Final Rejection of those claims.

IV. 35 U.S.C. § 103, Obviousness, Claims 9, 21 and 33

Claims 9, 21 and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over APA, Krishan, Nagahara and Kitagawa, and further in view of "Computer User's Dictionary" by Microsoft Press (hereinafter "Microsoft"). Microsoft is cited as teaching the definition of a "log file".

The Microsoft document, however, does not supply the deficiencies in the principal references, as described above, and claims 9, 21 and 33 should be allowable in their present form, at least, by virtue of their dependency from allowable claims, and it is respectfully requested that the Board reverse the Examiner's Final Rejection of those claims.



Gerald H. Glanzman
Reg. No. 25,035
YEE & ASSOCIATES, P.C.
PO Box 802333
Dallas, TX 75380
(972) 385-8777

CLAIMS APPENDIX

The text of the claims involved in the appeal are:

1. A method of updating firmware in a system component within a data processing system, the method comprising:

responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure, determining, by a service processor, whether the system component has a current level of the firmware; and

responsive to a determination that the system component does not have the current level of the firmware, updating a copy of the firmware stored in the system component in a background operation while the data processing system remains available to a user for other actions.

2. The method as recited in claim 1, further comprising:

responsive to a determination that the update failed, notifying a user of the update failure.

3. The method as recited in claim 1, wherein the system component is a system power control network card in an input/output drawer.

4. The method as recited in claim 1, wherein updating the copy of the firmware stored in the system component comprises transferring a current copy of the firmware stored in a non-volatile memory accessible by the service processor.

5. The method as recited in claim 4, wherein the non-volatile memory is a non-volatile random access memory.

6. The method as recited in claim 4, wherein the non-volatile memory is a flash memory.

7. A method for updating system firmware in a data processing system, the method comprising:

in the background, and responsive to receiving a notification that an operating system has been loaded following completion of an initialization procedure, determining whether a level of a firmware copy on a system component matches a current level of firmware stored on a non-volatile memory within the system; and

responsive to a determination that the level of the firmware copy is different from the current level, transferring the current level of firmware to the system component to update the firmware copy on the system component.

8. The method as recited in claim 7, further comprising:

determining, after the update, whether a new level of the firmware copy on the system component matches the current level of the firmware stored on the system memory; and

responsive to a determination that the new level does not match the current level, notifying a user of a firmware update failure.

9. The method as recited in claim 8, wherein the step of notifying the user of the firmware update failure comprises creating a log file.

10. The method as recited in claim 7, wherein the system component is a system power control network card within a input/output drawer.

11. The method as recited in claim 7, wherein the non-volatile memory is a non-volatile random access memory.

12. The method as recited in claim 7, wherein the non-volatile memory is a flash memory.

13. A computer program product in a computer readable media for use in a data processing system for updating firmware in a system component, the computer program product comprising:

first instructions, responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure, for determining, by a service processor, whether the system component has a current level of the firmware; and

second instructions, responsive to a determination that the system component does not have the current level of the firmware, for updating a copy of the firmware stored in the system component in a background operation while the data processing system remains available to a user for other actions.

14. The computer program product as recited in claim 13, further comprising:

third instructions, responsive to a determination that the update failed, for notifying a user of the update failure.

15. The computer program product as recited in claim 13, wherein the system component is a system power control network card in an input/output drawer.

16. The computer program product as recited in claim 13, wherein updating the copy of the firmware stored in the system component comprises transferring a current copy of the firmware stored in a non-volatile memory accessible by the service processor.

17. The computer program product as recited in claim 16, wherein the non-volatile memory is a non-volatile random access memory.

18. The computer program product as recited in claim 16, wherein the non-volatile memory is a flash memory.

19. A computer program product in a computer readable media for use in a data processing system for updating system firmware in a data processing system, the computer program product comprising:

first instructions, executed in the background, and responsive to receiving a notification that an operating system has been loaded following completion of an initialization procedure, for determining whether a level of a firmware copy on a system component matches a current level of firmware stored on a non-volatile memory within the system; and

second instructions, responsive to a determination that the level of the firmware copy is different from the current level, for transferring the current level of firmware to the system component to update the firmware copy on the system component.

20. The computer program product as recited in claim 19, further comprising:

third instructions for determining, after the update, whether a new level of the firmware copy on the system component matches the current level of the firmware stored on the system memory; and

fourth instructions, responsive to a determination that the new level does not match the current level, for notifying a user of a firmware update failure.

21. The computer program product as recited in claim 20, wherein notifying the user of the firmware update failure comprises creating a log file.

22. The computer program product as recited in claim 19, wherein the system component is a system power control network card within a input/output drawer.

23. The computer program product as recited in claim 19, wherein the non-volatile memory is a non-volatile random access memory.

24. The computer program product as recited in claim 19, wherein the non-volatile memory is a flash memory.

25. A system for updating firmware in a system component, the system comprising:

first means, responsive to receiving a notification that control has been transferred to a host operating system following completion of an initialization procedure, for determining, by a service processor, whether the system component has a current level of the firmware; and

second means, responsive to a determination that the system component does not have the current level of the firmware, for updating a copy of the firmware stored in the system component in a background operation while the data processing system remains available to a user for other actions.

26. The system as recited in claim 25, further comprising:

third means, responsive to a determination that the update failed, for notifying a user of the update failure.

27. The system as recited in claim 25, wherein the system component is a system power control network card in an input/output drawer.

28. The system as recited in claim 25, wherein updating the copy of the firmware stored in the system component comprises transferring a current copy of the firmware stored in a non-volatile memory accessible by the service processor.

29. The system as recited in claim 28, wherein the non-volatile memory is a non-volatile random access memory.

30. The system as recited in claim 28, wherein the non-volatile memory is a flash memory.

31. A system for updating system firmware in a data processing system, the system comprising:

first means, executed in the background, and responsive to receiving a notification that an operating system has been loaded following completion of an initialization procedure, for determining whether a level of a firmware copy on a system component matches a current level of firmware stored on a non-volatile memory within the system; and

second means, responsive to a determination that the level of the firmware copy is different from the current level, for transferring the current level of firmware to the system component to update the firmware copy on the system component.

32. The system as recited in claim 31, further comprising:

third means for determining, after the update, whether a new level of the firmware copy on the system component matches the current level of the firmware stored on the system memory; and

fourth means, responsive to a determination that the new level does not match the current level, for notifying a user of a firmware update failure.

33. The system as recited in claim 32, wherein notifying the user of the firmware update failure comprises creating a log file.

34. The system as recited in claim 31, wherein the system component is a system power control network card within a input/output drawer.

35. The system as recited in claim 31, wherein the non-volatile memory is a non-volatile random access memory.

36. The system as recited in claim 31, wherein the non-volatile memory is a flash memory.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.